

# SIGNÁLY A SYSTÉMY

2021/2022



## PROTOKOL K PROJEKTU

Band-stop filtr

## 4.1 Základy

```
close all; clear; clc;
```

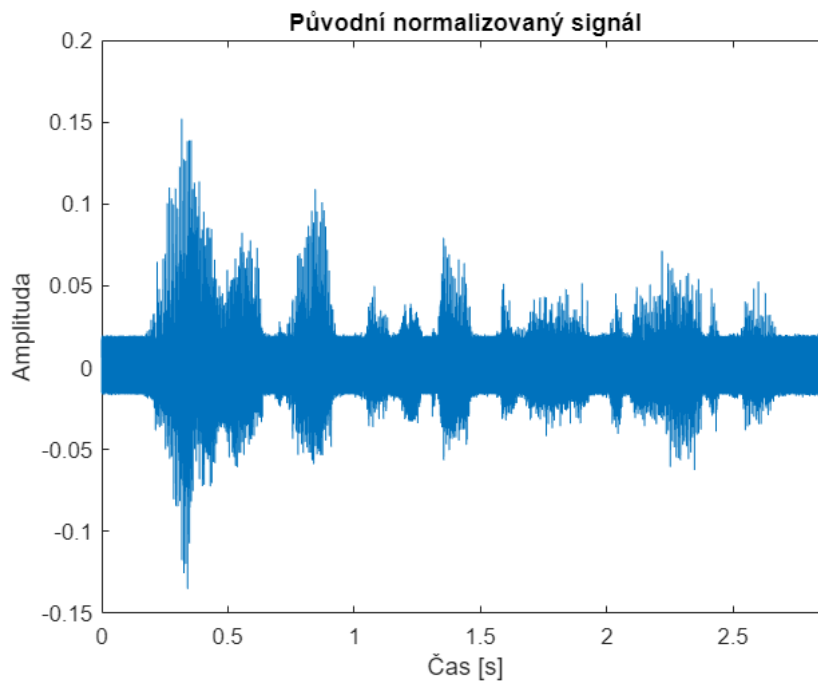
### Načtení zvuku

```
[audiodata, vzor_frekv] =  
audioread(['https://www.fit.vutbr.cz/study/courses/ISS' ...  
          '/public/proj2021-22/signals/xvalen29.wav']);  
  
delka = length(audiodata);  
delka_sek = length(audiodata) / vzor_frekv;  
MAX = max(abs(audiodata));  
MIN = min(abs(audiodata));  
mid = mean(audiodata);  
delka_vzorek = delka_sek / (delka - 1);  
t = 0 : delka_vzorek : delka_sek;
```

### Výpis vlastností

Vzorkovací frekvence = 16000 Hz  
Délka signálu ve vzorcích = 45978  
Délka signálu v sekundách = 2.8736 s  
Minimální hodnota normalizovaného signálu = 0  
Maximální hodnota normalizovaného signálu = 0.15204

### Zobrazení signálu



## 4.2 Předzpracování a rámce

```
pocet_ramcu = floor(delka / 512) - 1;
vzorku_ramec = 1024;
prekryti = 512;
znely_ramec = 11;
```

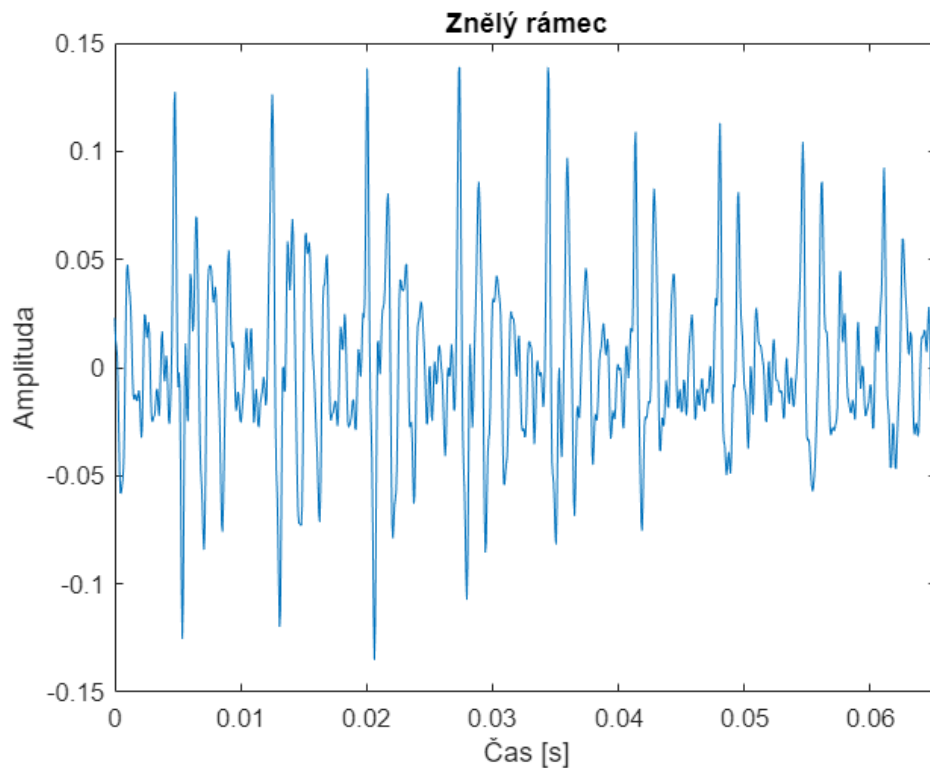
### Rozdělení do rámců

```
ramec = zeros(vzorku_ramec, pocet_ramcu);
for index = 0 : pocet_ramcu - 1
    ramec(:, index + 1) = reshape(audiodata( ...
        index * prekryti + 1 : index * prekryti + vzorku_ramec ...
    ), vzorku_ramec, []);
end
```

### Znělý rámeček s číslem 11

```
delka_ramec_sek = delka_sek / pocet_ramcu * 2;
t = 0 : delka_ramec_sek / (vzorku_ramec - 1) : delka_ramec_sek;
```

### Zobrazení znělého rámečku



## 4.3 DFT

### Vytvoření matice $\omega$

```
omega = exp(-2 * pi * 1i / vzorku_ramec);
n = 0 : vzorku_ramec - 1;
[xx, yy] = meshgrid(n);
W = omega .^ (xx .* yy);
```

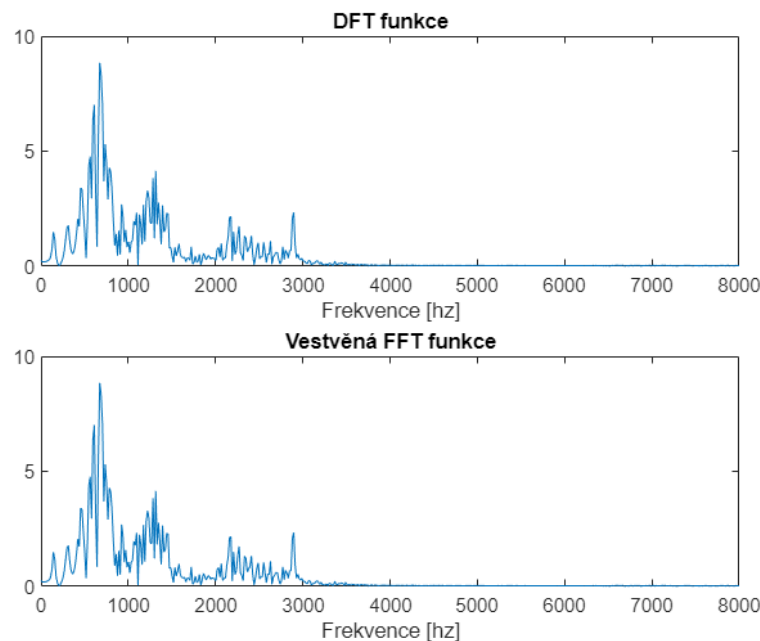
### DFT

```
DFT_vzorek = vzor_frekv / 2 / (vzorku_ramec / 2 - 1);
DFT = abs(W * ramec(:, znely_ramec));
DFT = DFT(1 : vzorku_ramec / 2);
t = 0 : DFT_vzorek : vzor_frekv / 2;
```

### Vestavěná FFT funkce

```
FFT = abs(fft(ramec(:, znely_ramec)));
FFT = FFT(1 : vzorku_ramec / 2);
frekvence = 0 : DFT_vzorek : vzor_frekv / 2;
```

### Porovnání DFT a FFT graficky

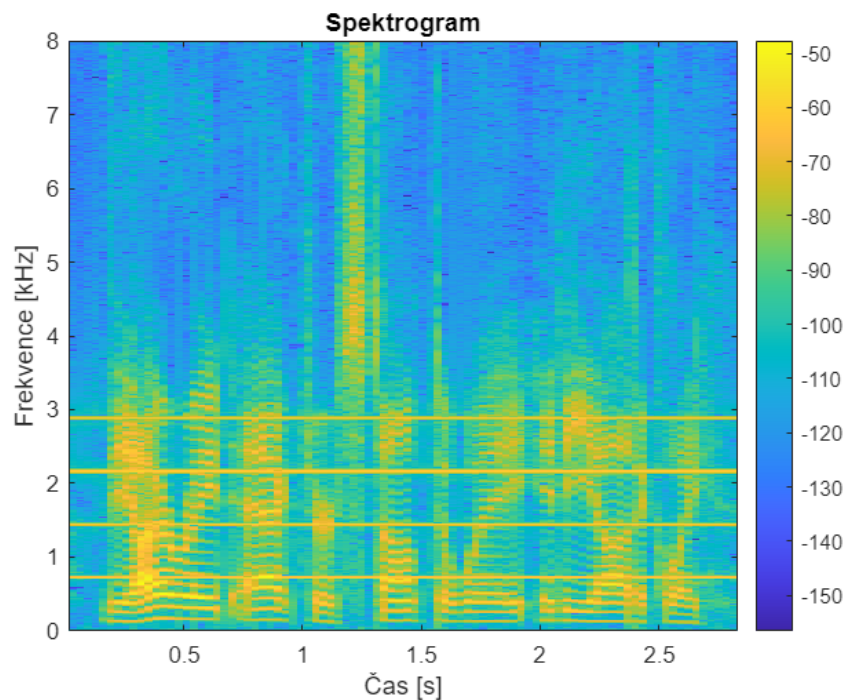


### Porovnání transformací číselně

```
avg_mean = mean(FFT - DFT);
total_sum = sum(abs(FFT - DFT));
```

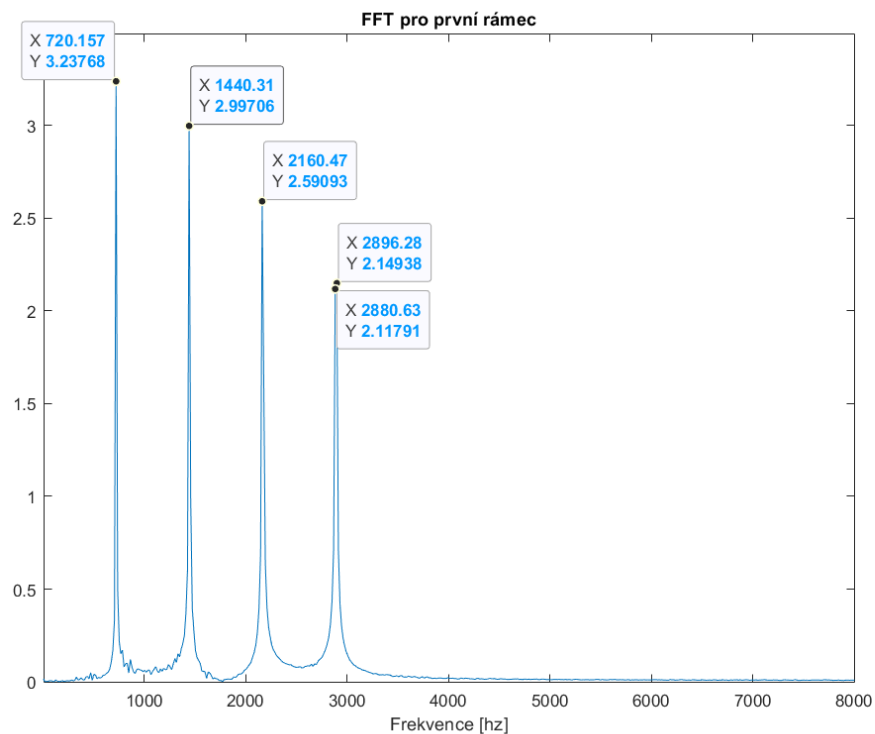
Průměrný rozdíl hodnot = 5.0272e-13  
Celkový rozdíl hodnot = 2.6443e-10

## 4.4 Spektrogram



## 4.5 Určení rušivých frekvencí

```
FFT = abs(fft(ramec(:, 1)));
FFT = FFT(1 : vzorku_ramec / 2);
```



## Určení největších hodnot a jejich indexů číselně

```
n = 1 : vzorku_ramec / 2;
FFT = [transpose(FFT); n];
sorted = sortrows(FFT.', 1).';
max_index = sort(sorted(2, end -3 : end));
max_index = frekvence(max_index);
```

### Čtvrtá hodnota nesedla přesně do DFT

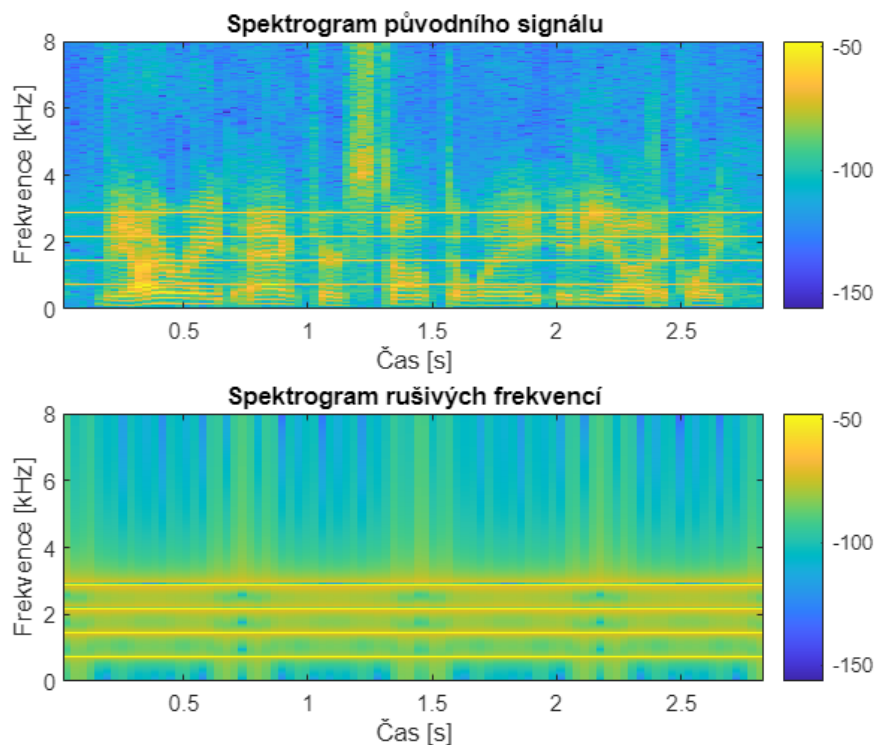
Rušivé frekvence =

720.16	1440.3	2160.5	2896.3
<b>720.16</b>	<b>1440.3</b>	<b>2160.5</b>	<b>2880.6</b>

## 4.6 Generování signálu

```
linear = linspace(0, delka_sek - delka_vzorek, vzor_frekv * delka_sek + 1);
linear(end) = [];
linear = sin(transpose(max_index * 2 * pi) * linear);
linear = sum(linear) / 4;
audiowrite("audio/4cos.wav", linear, vzor_frekv);
```

### Porovnání zvuků spektrogramem



## 4.7 Čistící filtr

```
zaverna_sirka = 30 / 2;
propustna_sirka = 50 + zaverna_sirka;
zvlneni = 3;
potlaceni = 40;
nyquist = vzor_frekv / 2;
B = zeros(4, 9);
A = zeros(4, 9);
H = dfilt.df2t.empty(0, 4);
```

### Vytvoření filtrů

```
for index = 1 : 4
    x = max_index(index);
```

### Band-stop butter filtr

```
h = fdesign.bandstop(x - propustna_sirka, x - zaverna_sirka, ...
    x + zaverna_sirka, x + propustna_sirka, zvlneni, potlaceni, ...
    zvlneni, vzor_frekv);
Hd = design(h, 'butter', 'MatchExactly', 'stopband');

sos = get(Hd, 'sosMatrix');
[b, a] = sos2tf(sos);
```

### Převedení do objektu

```
H(index) = dfilt.df2t(B(index, 1: end), A(index, 1 : end));
end
```

Koeficienty 1:

b = 1	-7.6847	26.146	-51.418	63.914	-51.418	26.146	-7.6847	1
a = 1	-7.5747	25.403	-49.245	60.342	-47.855	23.989	-6.9511	0.89178

Koeficienty 2:

b = 1	-6.7563	21.118	-39.544	48.375	-39.544	21.118	-6.7563	1
a = 1	-6.6674	20.566	-38.006	45.884	-37.018	19.511	-6.161	0.90005

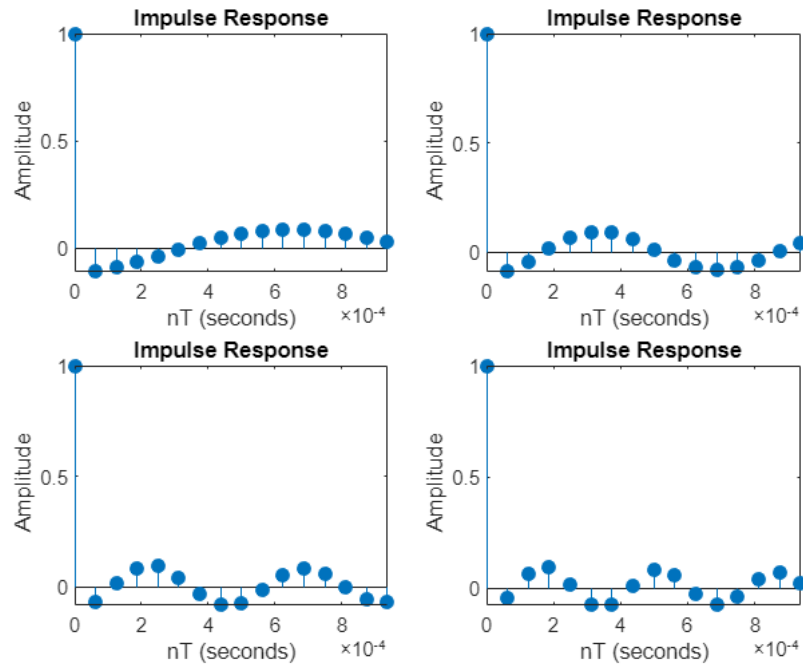
Koeficienty 3:

b = 1	-5.2911	14.498	-25.131	30.058	-25.131	14.498	-5.2911	1
a = 1	-5.2238	14.132	-24.185	28.56	-23.577	13.43	-4.8397	0.9032

Koeficienty 4:

b = 1	-3.4056	8.3492	-12.685	15.224	-12.685	8.3492	-3.4056	1
a = 1	-3.3631	8.141	-12.217	14.479	-11.916	7.7459	-3.1207	0.90508

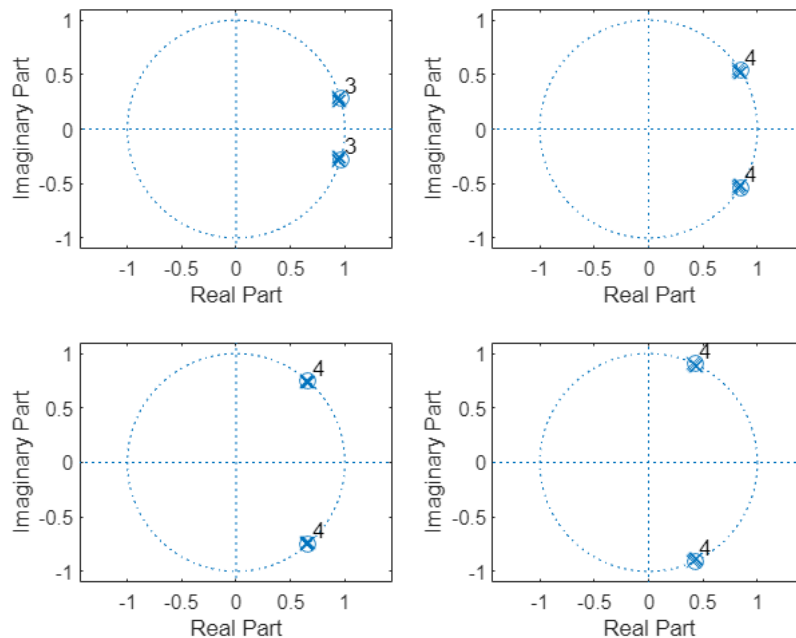
## Impulzní odezvy filtrů



## 4.8 Nulové body a póly

### Vypsání nulových bodů na Z - plane

```
b(1 : end) = B(index, 1 : end);
a(1 : end) = A(index, 1 : end);
zplane(b, a);
```

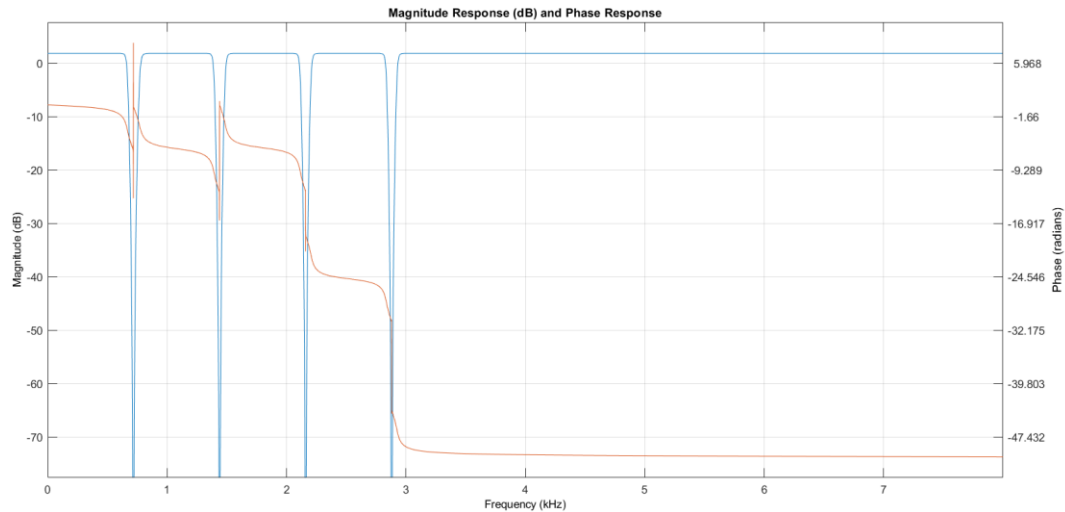




## 4.9 Frekvenční charakteristika

### Spojení filtrů do kaskády objektem

```
Hcas = dfilt.cascade(H);
freqz(Hcas, 2^18, vzor_frekv);
```



## 4.10 Filtrace

```
signal_filt = filter(Hcas, audiodata);
```

### Uložení audia

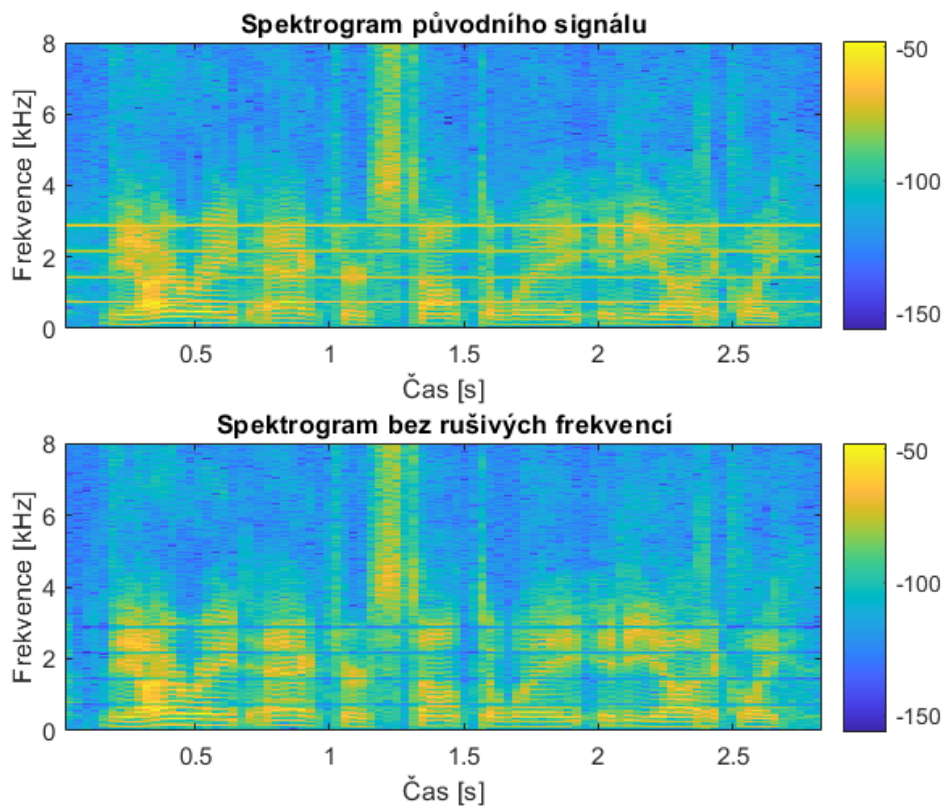
```
linear = linspace(0, delka_sek - delka_vzorek, vzor_frekv * delka_sek + 1);
linear(end) = [];
linear = sin(transpose(max_index * 2 * pi) * linear);
linear = sum(linear) / 4;
audiowrite("audio/clean_bandstop.wav", signal_filt, vzor_frekv);
```

### Porovnání audia poslechem

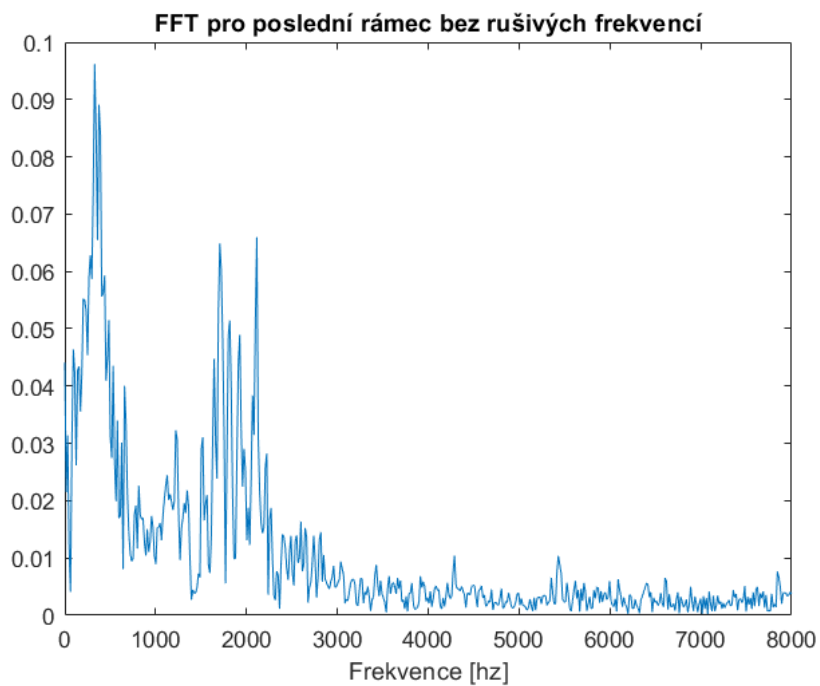
```
sound(audiodata, vzor_frekv);
pause(delka_sek + 1);
sound(signal_filt, vzor_frekv);
pause(delka_sek + 1);
```

## Závěrečná srovnání

### Porovnání spektrogramů



### FFT pro poslední rámeček



## Zdroje

### Vysvětlení pojmů

[https://en.wikipedia.org/wiki/DFT\\_matrix](https://en.wikipedia.org/wiki/DFT_matrix)  
<https://brilliant.org/wiki/discrete-fourier-transform/>  
[https://en.wikipedia.org/wiki/Butterworth\\_filter](https://en.wikipedia.org/wiki/Butterworth_filter)  
<https://stackoverflow.com/questions/5591278/high-pass-filtering-in-matlab>

### Funkce a ukázkové příklady

<https://www.mathworks.com/help/matlab/>  
<https://www.mathworks.com/help/signal/>  
<https://www.mathworks.com/matlabcentral/>

### Základy práce s programem Matlab

[https://www.mathworks.com/help/matlab/matlab\\_prog/create-live-scripts.html](https://www.mathworks.com/help/matlab/matlab_prog/create-live-scripts.html)  
<https://www.youtube.com/watch?v=VFt3UVw7VrE>  
[https://www.fit.vutbr.cz/study/courses/ISS/public/proj\\_studijni\\_etapa/1\\_matlab/](https://www.fit.vutbr.cz/study/courses/ISS/public/proj_studijni_etapa/1_matlab/)

# Obsah

## 4.1 ZÁKLADY

NAČTENÍ ZVUKU

## 4.2 PŘEDZPRACOVÁNÍ A RÁMCE

ROZDĚLENÍ DO RÁMCŮ

ZNĚLÝ RÁMEC S ČÍSLEM 11

ZOBRAZENÍ ZNĚLÉHO RÁMCE

## 4.3 DFT

VYTVOŘENÍ MATICE  $\Omega$

DFT

VESTAVĚNÁ FFT FUNKCE

POROVNÁNÍ DFT A FFT GRAFICKY

POROVNÁNÍ TRANSFORMACÍ ČÍSELNĚ

## 4.4 SPEKTROGRAM

## 4.5 URČENÍ RUŠIVÝCH FREKVENCÍ

URČENÍ NEJVĚTŠÍCH HODNOT A JEJICH INDEXŮ ČÍSELNĚ

ČTVRTÁ HODNOTA NESEDLA PŘESNĚ DO DFT

## 4.6 GENEROVÁNÍ SIGNÁLU

POROVNÁNÍ ZVUKŮ SPEKTROGRAMEM

## 4.7 ČISTÍCÍ FILTR

VYTVOŘENÍ FILTRŮ

BAND-STOP BUTTER FILTR

PŘEVEDENÍ DO OBJEKTU

IMPULZNÍ ODEZVY FILTRŮ

## 4.8 NULOVÉ BODY A PÓLY

VYPSÁNÍ NULOVÝCH BODŮ NA  $Z$  - PLANE

## 4.9 FREKVENČNÍ CHARAKTERISTIKA

SPOJENÍ FILTRŮ DO KASKÁDY OBJEKTEM

## 4.10 FILTRACE

ULOŽENÍ AUDIA

POROVNÁNÍ AUDIA POSLECHEM

## ZÁVĚREČNÁ SROVNÁNÍ

POROVNÁNÍ SPEKTROGRAMŮ

FFT PRO POSLEDNÍ RÁMEC

## ZDROJE

VYSVĚTLENÍ POJMŮ

FUNKCE A UKÁZKOVÉ PŘÍKLADY

ZÁKLADY PRÁCE S PROGRAMEM MATLAB